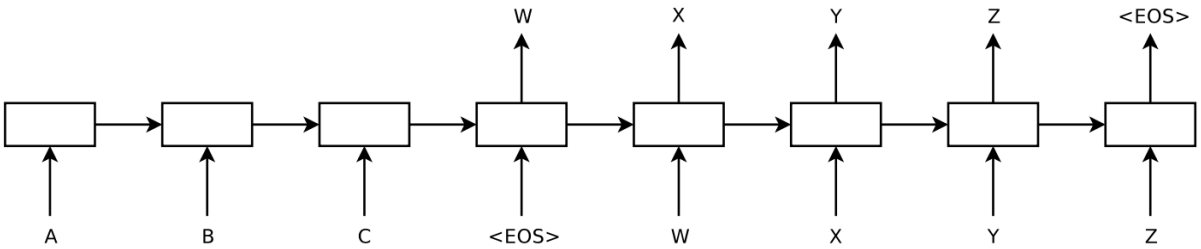
Seq2Seq

Abstract

- DNNs는 대규모의 라벨링된 학습 데이터가 주어졌을 때만 효과적으로 작동, 시퀀스를 시퀀스로 매핑하는 데는 사용할 수 없음
- 논문에서는 시퀀스 구조에 대한 최소한의 가정만을 적용하면 end to end 시퀀스 학습 접근법 제시
- LSTM을 사용하여 입력 시퀀스를 고정된 차원의 벡터로 매핑 후, 또 다른 심층 LSTM을 이용하여 해당 벡터에서 target sentence 디코딩
- source sentence의 단어 순서를 뒤집어서 LSTM의 성능 향상시킴
  -> source, target sentence의 단기 의존성을 증가시켜 최적화 문제를 쉽게 만듦

1. Introduction

- DNNs은 음성 인식 및 시각적 객체 인식과 같은 어려운 문제에서 뛰어난 성능을 갖고 있는 문제
- 두 개의 은닉층을 사용하여 N개의 N-비트 숫자를 정렬할 수 있음
  -> 복잡한 연산을 학습할 수 있음
- 라벨링된 학습 데이터가 주어지면, DNN은 지도학습 기반의 역전파 알고리즘 통해 학습 가능
- 입력과 타겟이 고정된 차원의 벡터로만 표현 가능하다는 한계로, 입력과 출력의 차원을 사전에 알고 고정해야 함
- LSTM 구조를 적용하여 sequence to sequence 문제 해결 가능



-입력 문장 "ABC"를 읽고 출력 문장 "WXYZ"를 생성

- source sentence의 단어 순서를 반대로 배치하고, 타겟 문장은 그대로 유지하는 방식으로 학습 및 테스트 진행
  -> 데이터 내에서 많은 단기 의존성을 유도하여 최적화 문제 단순하게 만듦, 확률적 경사 하강법이 긴 문장에서도 문제없이 학습
- LSTM은 가변 길이의 입력 문장을 고정 차원의 벡터 표현으로 변환 가능
- 번역 학습 과정에서 LSTM은 의미적으로 유사한 문장을 가까운 벡터 공간에 배치, 다른 문장은 멀리 떨어뜨림

2. The model

RNN은 피드포워드 신경망을 순차데이터(sequence)로 일반화한 모델

- RNN은 입력과 출력이 미리 정해진 정렬 관계를 가지면 쉽게 시퀀스를 시퀀스로 매핑할 수 있음
- 하지만 문제점은 입력과 출력 시퀀스의 길이가 서로 다르고 복잡한 관계인 방법에서는 명확하지 않음
- 일반적인 시퀀스 학습을 위한 가장 간단한 전략은 하나의 RNN을 사용하여 입력 시퀀스를 고정 크기의 벡터로 변환한 후, 또 다른 RNN을 사용하여 이 벡터를 목표 시퀀스로 변환하는 것
  -> 그러나 이 방법은 장기 의존성이 발생하여 RNN을 학습하는 게 어려워짐

LSTM은 장기 의존성을 가진 문제를 학습할 수 있어, 이런 설정에서도 성공할 가능성이 있음

- LSTM의 목표는 서로 다른 입력 시퀀스와 출력 시퀀스의 조건부 확률p(y1,...,yT′ ∣ x1,...,xT)p(y*1, ..., y*{T'} | x_1, ..., x_T)p(y1,...,yT′ ∣ x1,...,xT)을 추정하는 것
- 입력 시퀀스와 출력 시퀀스의 길이가 다를 수 있음
- 입력 시퀀스의 마지막 은닉 상태를 고정 차원의 벡터로 변환한 후, 이를 초기 상태로 설정하여 출력 시퀀스의 확률을 계산할 수 있음, 각 확률 분포는 어휘 전체에 대한 소프트맥스로 표현됨
- 모든 문장이 특별한 문장 종료 토큰 로 끝나도록 요구함 -> 모델이 가능한 모든 길이의 시퀀스에 대한 확률 분포를 정의

Seq2seq 모델은 LSTM과 세 가지 차이점 존재

1. 입력 시퀀스를 처리하는 LSTM과 출력 시퀀스를 생성하는 LSTM을 각각 따로 사용
   -> 계산 비용이 거의 증가하지 않으면서 모델의 파라미터 수를 늘릴 수 있음
   -> 여러 언어 쌍을 동시에 학습하는 것이 자연스러워짐
2. 얕은 LSTM보다 깊은 LSTM이 우수한 성능을 보였기 때문에 4개 층의 LSTM 선택
3. 입력 문장의 단어 순설르 역순으로 바꾸는 것이 효과적이라는 점을 발견
   -> 예를 들어, 문장 "a, b, c"를 "α, β, γ"로 매핑하는 대신, "c, b, a"를 "α, β, γ"로 매핑하도록 학습
   -> "a"가 "α"와 가까운 위치에 오고, "b"가 "β"와 비교적 가까운 위치에 오게 됨
   -> 확률적 경사 하강법(SGD)이 입력과 출력 사이의 관계를 더 쉽게 학습하도록 도움

3. Experiments

WMT'14 English to French 데이터셋에 두 가지 방식으로 논문의 방법 적용

1. SMT 시스템 없이 입력 문장을 직접 번역
2. SMT 기반 시스템의 n-best 리스트를 다시 정수화하는 방식

3.2 Decoding and Rescoring

- 실험의 핵심은 다수의 문장 쌍을 학습하는 대규모 심층 LSTM을 훈련하는 것
- 최적의 번역을 찾기 위해 좌에서 우로 진행하는 빔 서치(beam search) 디코더를 사용

- 디코더는 B개의 부분적 가설(partial hypothesis)을 유지하는데, 부분적 가설이란 번역의 일부(접두사)를 의미
- 각 time step에서 빔 내의 모든 부분적 가설을 가능한 모든 단어로 확장
- 확장된 가설 중 가장 가능성이 있는 번역이 log 확률에 따라 선택되며, 나머지는 버려짐

3.3 Reversing the Source Sentences

- source sentence를 역순으로 변환하면, 소스와 타겟 언어에서 대응하는 단어들 간의 평균 거리는 변하지 않지만, 소스 문장의 처음 몇 단어가 타겟 문장의 처음 몇 단어와 훨씬 가까워짐
  -> 최소 시간 지연이 감소하여, 역전파 과정에서 소스와 타겟 문장 간의 의사소통이 더 쉬워지고 전체적인 성능 향상
- 역순 변환된 소스 문장으로 훈련된 LSTM이 원본 소스 문장으로 훈련된 LSTM보다 긴 문장에서도 더 우수한 성능

5. Conclusion

- 대형 심층 LSTM 모델이 어휘 크기에 제한이 없는 기존 통계적 기계 번역(SMT) 시스템보다 대규모 기계 번역(MT) 작업에서 더 우수한 성능을 발휘할 수 있음을 보임
  -> 충분한 학습 데이터가 제공될 경우 다른 시퀀스 학습 문제에서도 좋은 성능을 발휘할 가능성이 높음을 시사함
- 학습 문제를 단순하게 만들기 위해서는 단기 의존성이 많은 형태로 문제를 인코딩하는 것이 중요함
- 역순 데이터로 학습된 LSTM 모델은 긴 문장 번역에서도 좋은 성능을 보임
- 논문의 방법이 단순하고 최적화가 덜 된 상태에서도 기존 SMT 시스템을 능가함